# Chapter 10

# File Specification and Naming

The File Specification and Naming Standard defines the PDS conventions for forming file specifications and file names. This standard is based on Level 1 and Level 2 of the international standard ISO 9660, “Information Processing - Volume and File Structure of CD-ROM for Information Interchange.”

***ISO 9660 Level 1 versus ISO 9660 Level 2***

- PDS recommends that archive products comply with the ISO 9660 Level 1 specification. Specifically, PDS recommends that CD-ROM volumes which are anticipated to have a wide distribution use file identifiers consisting of a maximum of 8 characters for the file name and 3 characters for the extension to comply with the ISO 9660 Level 1 specification.

- PDS has adopted the ISO 9660 Level 2 specification for those archive products for which there are compelling reasons to relax the 8.3 filename restrictions.

## 10.1 File Specification Standards

A file specification consists of the following elements:

| | |
|---|---|
| A complete directory path name | (as discussed in the *Directory Types and Naming* chapter of this document) |
| A file name having an extension | |

The PDS has adopted the UNIX/POSIX forward slash operator (/) for use in path names. Directory path name formation is discussed further in the *Directory Types and Naming* chapter of this document.

The following is an example of a simple file specification. The file specification identifies the location of the file relative to the root of a volume, inlcuding the directory path name.

| | |
|---|---|
| File Name: | TG15N122.IMG |
| File Specification: | TG15NXXX/TG15N1XX/TG15N12X/TG15N122.IMG |

Do not use path or file names that correspond to operating system specific names, such as:

AUX COM1 CON LPT1 NUL PRN

### 10.1.1 ISO 9660 Level 1 Specification

A file name consists of a basename and an extension separated by a required FULL STOP (a.k.a. period) character (.). The total length of the file name shall not exceed 12 characters. The length of the base name shall not exceed 8 characters and the extension shall not exceed 3 characters. The file identifier must be suffixed with a version number consisting of a semicolon and an integer to comply with the ISO 9660 Level 1 specification. Both the base name and extension shall contain only the upper case alphanumeric character set (A- Z, 0-9), and underscore (_). These requirements are often referred to as the 8.3 (8 dot 3) file naming convention. These limitations exist primarily to accommodate older computer systems (e.g. IBM DOS-based PCs) that cannot handle longer file names. Since PDS archive volumes are designed to be read on many platforms, including PCs, these restrictions are necessary.

Preferred format: FILENAME (1..8 characters) "." EXTENSION (3 characters)

Allowable format: FILENAME (1..8 characters) "." EXTENSION (1..3 characters)

Actual format
on archive media: FILENAME (1..8 characters) "." EXTENSION (1..3 characters) ";1"

### 10.1.2 ISO 9660 Level 2 Specification

A file name consists of a basename and an extension separated by a required FULL STOP (a.k.a. period) character (.). The total length of the file name shall not exceed 31 characters. Both the base name and extension shall contain only the upper case alphanumeric character set (A- Z, 0-9), and underscore (_). These requirements are often referred to as the 27.3 (27 dot 3) file naming convention. The file identifier must be suffixed with a version number consisting of a semicolon and an integer to comply with the ISO 9660 Level 2 specification. It is strongly recommended that the PDS file naming recommendations for file extensions be followed (including the use of the standard 3 character extensions).

Preferred format: FILENAME (1..27 characters) "." EXTENSION (3 characters)

Allowable format: FILENAME (1..29 characters) "." EXTENSION (1..29 characters)

Actual format
on archive media: FILENAME (1..29 characters) "." EXTENSION (1..29 characters) ";1"

## 10.2 File Naming Standards

The following sections identify the PDS required and reserved file names and file extensions. Required and reserved file names and extensions provide consistency across PDS archive volumes, which is helpful to users. Also, software tools can make use of this predictability.

Required means that if a file contains a given type of information, it shall have the given name or

extension. Reserved means that if a file has a given name or extension, it shall contain that type of information. For example, the volume object is contained in, and only in, the file named VOLDESC.CAT. It is a required file name. A file named TG15N122.IMG contains an image. File extensions should be used to identify the data type of a file. This is reflected in the required and reserved file extensions listed later in this chapter.

### 10.2.1 Required File Names

VOLDESC.CAT - This file name must be used for the file containing the volume object. This required file is placed in the ROOT directory of a volume.

*objectname*.CAT - This category of file name must be used for files containing a catalog object. These files, if present on a volume, must be placed in the CATALOG directory of a volume. The Software Inventory catalog object may also be placed in the SOFTWARE hierarchy under the appropriate DOC directory.

"*objectname*" is one of the commonly used catalog objects listed below. The form of the file name varies if one or more objects are included in the archive product. For example, if a volume contains a single data set, the data set object shall be contained in the file named DATASET.CAT. If the volume contains multiple data sets and the data set objects are contained in separate files, each file shall be named xxxxxxDS.CAT where "xxxxxx" is replaced with an acronym of up to six characters for the data set.

It is possible to include all of the catalog objects in a single file. However, PDS strongly discourages this approach. One of the advantages to supplying individual files is that each catalog object can be 'plugged into' other archive products. If a single file is used to contain all catalog objects, it must be named CATALOG.CAT. The pointer expression becomes:

^CATALOG = "CATALOG.CAT"

If catalog objects are organized in separate files or sets of files, pointer expressions shall be constructed according to the following table. Under "File Name", the first line shows the file name to be used if a single catalog file is present on the volume for the particular type of catalog object named. The second shows the syntax and file name convention to be followed if multiple catalog files are present for the named object.

| Catalog Pointer Name | | File Name |
|---|---|---|
| ^DATA_SET_CATALOG | = | "DATASET.CAT" |
| | = | {"xxxxxxDS.CAT","yyyyyyDS.CAT"} |
| ^DATA_SET_COLLECTION_CATALOG | = | "DSCOLL.CAT" |
| | = | {"xxxxxDSC.CAT","yyyyyDSC.CAT"} |
| ^DATA_SET_MAP_PROJECTION_CATALOG | = | "DSMAP.CAT" |
| | = | {"xxxDSMAP.CAT","yyyDSMAP.CAT"} |
| ^INSTRUMENT_CATALOG | = | "INST.CAT" |
| | = | {"xxxxINST.CAT","yyyyINST.CAT"} |
| ^INSTRUMENT_HOST_CATALOG | = | "INSTHOST.CAT" |
| | = | {"xxxxHOST.CAT","yyyyHOST.CAT"} |

```
^MISSION_CATALOG              =  "MISSION.CAT"
                              =  {"xxxxxMSN.CAT","yyyyyMSN.CAT"}
^PERSONNEL_CATALOG            =  "PERSON.CAT"
                              =  {"xxxxPERS.CAT,"yyyyPERS.CAT"}
^REFERENCE_CATALOG            =  "REF.CAT"
                              =  {"xxxxxREF.CAT","yyyyyREF.CAT"}
^SOFTWARE_INVENTORY_CATALOG   =  "SWINV.CAT"
                              =  {"xxxSWINV.CAT", "yyySWINV.CAT"}
^TARGET_CATALOG               =  "TARGET.CAT"
                              =  {"xxxTGT.CAT", "yyyTGT.CAT"}
```

AAREADME.TXT- This file name must be used for the file that contains a terse description of the volume contents. This required file is placed in the ROOT directory of a volume.

ERRATA.TXT- This file name is used for a file used to provide comments as well as to report errors. Cumulative comments for a volume set are kept in this file (although cumulative comments are optional for a volume set). This optional file is placed in the ROOT directory of a volume.

VOLINFO.TXT - This file name must be used for the file containing detailed information necessary to interpret the data set(s) contained on the volume. When present, this file is placed in the DOCUMENT directory of a volume. The VOLINFO.TXT file is referenced in the catalog object as ^DESCRIPTION = "VOLINFO.TXT".

NOTE: PDS strongly discourages the use of the VOLINFO.TXT file. This file can be provided as an alternate for individual catalog objects, but this approach negates the ability to re-use each catalog object in other archive products. PDS requires that either the VOLINFO.TXT file, or the *objectname.*CAT files described above, be present on the volume.

The following xxINFO.TXT files are required to appear in the non-data subdirectories that appear on the volume:

| Sub-directory | File |
|---|---|
| BROWSE | BROWINFO.TXT |
| CALIB | CALINFO.TXT |
| CATALOG | CATINFO.TXT |
| DOCUMENT | DOCINFO.TXT |
| EXTRAS | EXTRINFO.TXT |
| GAZETTER | GAZINFO.TXT |
| GEOMETRY | GEOMINFO.TXT |
| INDEX | INDXINFO.TXT |
| LABEL | LABINFO.TXT |
| SOFTWARE | SOFTINFO.TXT |

The following xxINFO.TXT files are recommended in appropriate SOFTWARE subdirectories:

| **Sub-directory** | **File** |
|---|---|
| SOFTWARE/PC | PCINFO.TXT |
| SOFTWARE/MAC | MACINFO.TXT |
| SOFTWARE/SUN | SUNINFO.TXT |
| SOFTWARE/SGI | SGIINFO.TXT |

The following file names should be used for INDEX files:

| | Single Index | Multi-Index |
|---|---|---|
| Single Volume | INDEX.TAB | axxINDEX.TAB |
| Multi-volume set | INDEX.TAB and CUMINDEX.TAB | axxINDEX.TAB and axxCMIDX.TAB |

### 10.2.2 Reserved File Names

VOLDESC. SFD - for use with a file containing an SFDU Reference Class object for an archive volume. Note that this file is identified here for backward compatability with previous versions of the PDS standards and is not to be used in current archive products.

### 10.2.3 Required File Extensions

.CAT - for use with a file containing a catalog object

.FMT - for use with an include file containing structural information (meta data) describing a data object

.LBL - for use with a file containing a detached PDS label for any class of data object. Note that a file containing a detached label should have the same base name as its associated data file, but the extension .LBL

.TXT - for use with a file described by the TEXT data object

.ASC - for use with a file containing a document in ASCII text format described by a label containing a DOCUMENT object definition.

### 10.2.4 Reserved File Extensions

| Extension | Description (for use with a file containing) |
|---|---|
| DAT | Binary files (other than images) |
| TAB | Table data (Note: this extension is also used for table data in ASCII form described by a detached PDS label) |
| IMG | Image data |
| IBG | Browse image data |
| IMQ | Image data that have been compressed |
| HTM | HTML document |
| TEX | TeX or LaTeX document |
| PDF | Adobe PDF document |
| DOC | Microsoft Word document |
| RTF | Rich Text document |
| PS | Postscript |
| EPS | Encapsulated Postscript |
| GIF | GIF image |
| JPG | JPEG image |
| QUB | Spectral (or other) image qubes |
| XSP | SPICE Transfer format SPK (ephemeris) files |
| BSP | SPICE Binary format SPK (ephemeris) files |
| XC | SPICE Transfer format CK (pointing) files |
| BC | SPICE Binary format CK (pointing) files |
| TI | SPICE Text IK (instrument parameters) files |
| TLS | SPICE Leapseconds kernel files |
| TPC | SPICE Physical and cartographic constants kernel files |
| TSC | SPICE Spacecraft clock coefficients kernel files |
| XES | SPICE E-kernel files. |
| EXE | Application or Executable |
| MAK | Makefile for compiling / linking application or executable |
| OBJ | Object file |
| DLL | Dynamic Link Library |
| LIB | Library of object files |
| ZIP | Zip-compressed files within PDS. |

NOTE: Additional file extensions are reserved for use for document files only and are described in the *Documentations* chapter in this document.

## 10.3 File Naming Guidelines

In cases where file names will contain an identification value constructed from the time tag or data object identifier, the following forms are suggested (but not required):

Pnnnnnnn.EXT

where P is one of the following:
C - The following value is a clock count value (C3345678.IMG)
T - The following value is a time value (T870315.TAB)
F - The following value is a FrameID or an ImageID (F242AO3.IMG)
N - The following value is a numeric file identification number (N003.TAB).